Semi-automatic Segmentation of Scattered and Distributed Objects

Semi-automatic Segmentation of Scattered and Distributed Objects

Muhammad Shahid Farid[1,2⋆], Maurizio Lucenteforte[2], Muhammad Hassan Khan[3], and Marco Grangetto[2]

[1] College of Information Technology, University of the Punjab, Lahore, Pakistan
[2] Dipartimento di Informatica, Università degli Studi di Torino, Torino, Italy
[3] Pattern Recognition Group, University of Siegen, Siegen, Germany

Abstract. This paper presents a novel object segmentation technique to extract objects that are potentially scattered or distributed over the whole image. The goal of the proposed approach is to achieve accurate segmentation with minimum and easy user assistance. The user provides input in the form of few mouse clicks on the target object which are used to characterize its statistical properties using Gaussian mixture model. This model determines the primary segmentation of the object which is refined by performing morphological operations to reduce the false positives. We observe that the boundary pixels of the target object are potentially misclassified. To obtain an accurate segmentation, we recast our objective as a graph partitioning problem which is solved using the graph cut technique. The proposed technique is tested on several images to segment various types of distributed objects e.g. fences, railings, flowers. We also show some remote sensing application examples, i.e. segmentation of roads, rivers, etc. from aerial images. The obtained results show the effectiveness of the proposed technique.

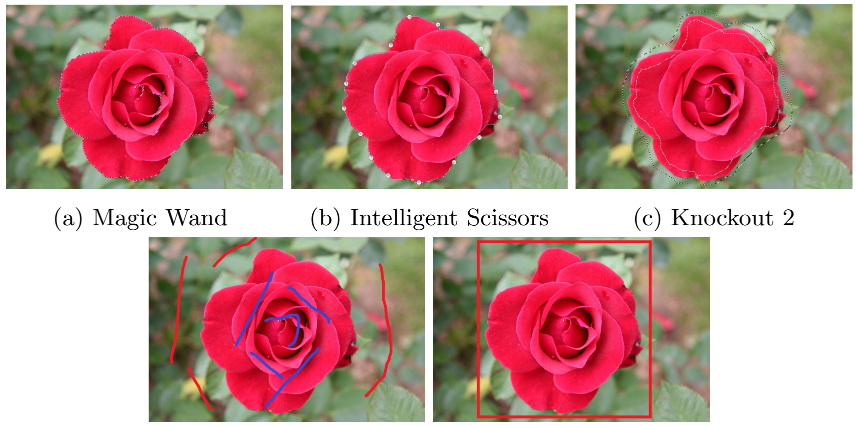
Keywords: Object segmentation, Gaussian mixture model, Graph-cuts

1 Introduction

Object segmentation refers to the extraction of a particular object from an image. It is a binary pixel labeling problem that partitions the image into two regions; foreground and background. Various interactive solutions to this problem have been proposed in literature, e.g. [1–7] and that usually require human user assistance to obtain satisfactory results. User assistance in object segmentation is used to guide the segmentation process and it is usually provided in the form of few scribbles on the target object and on the background. Each segmentation approach requires a different level of user assistance to obtain a neat segmentation.

Magic Wand and Lasso tool [8] provided by many image editing tools are considered to be the oldest and the simplest object segmentation techniques.

⋆ The major part of this research was done when the author was associated with institute[2].



(a) Magic Wand (b) Intelligent Scissors (c) Knockout 2

(d) Graph Cut, Lazy Snapping (e) GrabCut

Fig. 1: User assistance required in existing well-known object segmentation techniques.

Their accuracy totally depends on the user expertise and usually it takes a huge amount of time to obtain an accurate segmentation. Intelligent scissors [9] is an interactive segmentation tool which exploits the object contours to allows quick and accurate object segmentation. The user provides few seeds on the object boundary and it uses 'live-wire' boundary detection [10] to find the optimal contour between the seed points. Its results are usually satisfactory with adequate user assistance. Many other objects contour based segmentation methods have also been proposed e.g., [11–13]. Corel Knockout 2 [14] allows the user to specify the object trimap with the pencil tool and the pixel membership of the unknown region to foreground or background class is decided by computing the ratio of the distance between of the pixel color to the object color and background color.

Graph based segmentation [15] has gained widespread popularity due to its superior performance. In graph based segmentation approaches, the image is represented as weighted undirected graph and the segmentation is solved by finding global minimum of the energy function defined over the graph. According to [16], a considerable user effort is required to obtain satisfactory results. A review of well-known graph based image segmentation techniques can be found in [17]. Lazy Snapping [18] is another interactive segmentation which requires the user to mark few lines on the target object and the background. The segmentation is performed through graph cut technique [15]. It shows adequate results, however the user is fatigued in terms of post processing boundary editing which is quite similar to the seeding step in Intelligent scissors method.

GrabCut [16] is an iterative solution to graph cut optimization based on initial hard segmentation. Border matting [19] is used to obtain fine segmentation of object boundaries. GrabCut requires the user to draw a rectangle around the target object, the region outside of this rectangle is treated as background. The foreground and the background are represented with Gaussian mixture models (GMM) and the segmentation is obtained through global optimization which can be improved interactively. GrabCut is easy to use, however to refine the segmentation the user may need to feedback the system. The segmentation quality of GrabCut is significantly better than its ancestors. Other well known graph based segmentation approaches are [17, 20–24]. A comprehensive description of energy functions that can be minimized using graph cuts is given in [25] and an overview of graph cuts may be found in [26]. Fig. 1 shows the user assistance required in the interactive segmentation techniques described above.

Most existing segmentation techniques take trimap as input and are capable to accurately segment a single coherent object. However, these techniques are not effective in segmenting distributed objects which may cover the entire image, e.g. fences, roads, rivers, railings. Indeed, in such cases the definition of the trimap turns to be a very time consuming, tiresome and error prone task. In this paper a novel object segmentation technique is presented to extract objects that are spread on the whole image. The main contributions are:

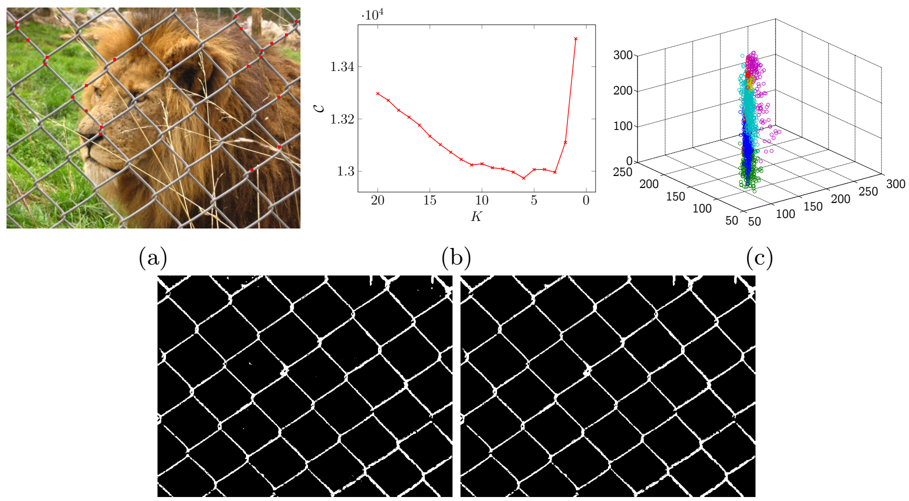
- Proposal of a novel segmentation technique to extract distributed or scattered objects;
- Minimal and easy user assistance in the form of few mouse clicks on the target object is required;
- The undesired background is automatically inferred without user intervention and Gaussian mixtures are exploited for primary segmentation;
- A trimap is created automatically and then graph based segmentation is used as the last refinement step;

The effectiveness of the proposed technique is tested on heterogeneous images to segment various types of objects. Moreover, its application in remote sensing to segment the roads, streams and rivers from aerial images is also presented.

2 Proposed Object Segmentation Technique

The proposed object segmentation technique is a multi-stage algorithm that takes the user input in the form of few mouse clicks on the target object. Based on this input the color characteristics the target object are estimated using mixture of Gaussians and a rough segmentation is obtained which is then refined through simple morphological operations. In the final stage the fine segmentation is achieved by recasting the problem as graph partition and solving it trough graph-cuts.

Let I be an input image and let P be the n points marked by the user on the target object. To achieve better modeling accuracy the sample data is increased by including the κ-neighboring points around each $p_i \in P$, defined as $\kappa \times \kappa$ square matrix centered at p_i, with κ odd. It turns out that a total of $\kappa^2 n$ points representative of the target objects are collected. Each pixel is represented with its red, green and blue components in RGB color space.



(a) (b) (c)

(d) (e)

Fig. 2: (a) Image with user input (red squares), (b) Model order estimation through MDL method with EM algorithm, (c) Cluster wise data distribution in the estimated optimal order GMM, (d) The initial segmentation map (zoom in to see the isolated blobs appeared as a result of false positives), (e) Segmentation map after false positive elimination.

2.1 Estimating Gaussian Mixture Models

Gaussian Mixture Model (GMM) is considered to be a quite general and accurate approach to represent the statistical properties of a data. A GMM is a parametric probabilistic model given by a weighted summation of Gaussian density functions. In particular, the likelihood of a pixel x, given the Gaussian mixture model $\mathcal{G}$ is defined as:

$$p(\mathrm{x}|\mathcal{G}) = \sum_{i=1}^{K} \pi_i g(\mathrm{x} \mid \mu_i, \Sigma_i) \tag{1}$$

where K is the number of Gaussian components, π_i, μ_i and Σ_i are the weight, mean and covariance of i-th Gaussian component with

$$g(\mathrm{x} \mid \mu_i, \Sigma_i) = \frac{1}{\sqrt{(2\pi)^3 \mid \Sigma_i|}} \exp(-\frac{1}{2}(\mathrm{x} - \mu_i)^\top \Sigma_i^{-1}(\mathrm{x} - \mu_i)) \tag{2}$$

The model parameters $\{\pi_{1,\cdots,K}, \mu_{1,\cdots,K}, \Sigma_{1,\cdots,K}\}$ can be estimated through expectation maximization (EM) algorithm [27, 28].

The only limitation of GMM and almost all other clustering approaches is to specify the number of clusters K to use for modeling. To this end, we exploit the Minimum Description Length (MDL) method [29] to estimate the the model order. The MDL principal selects the model that results in minimum code length of the data and the model parameters. Indeed, it attempts to minimize the following function $\mathcal{C}$:

$$\mathcal{C}(\mathcal{G}, K) = -\log p(\mathrm{x} \mid \mathcal{G}) + \frac{1}{2} L \log(M) \tag{3}$$

where L is the code length of model parameters $\mathcal{G}$. M is the size of the data i.e., $M = 3\kappa^2 n$ as there are $\kappa^2 n$ points and each is represented with 3 values. The minimum value of $\mathcal{C}$ in Eq. 3 is achieved by minimizing the log-likelihood term (increasing the maximum-likelihood (ML)) and the code length required to describe the model parameters and the data. Starting with K_0, a fixed value greater than the number of expected components, EM algorithm is used to estimate $\mathcal{G}$ and MDL is used to find the optimal order. After each iteration, the two most similar clusters are merged and the process is repeated until K=1. Finally, $\mathcal{G}$ and K corresponding to smallest MDL value $\mathcal{C}$ are selected as optimal model parameters.

2.2 Initial Segmentation and Rectification

The Gaussian Mixture Model $\mathcal{G}$ is used to obtain the raw segmentation of the target object using Mahalanobis distance [30], computed as:

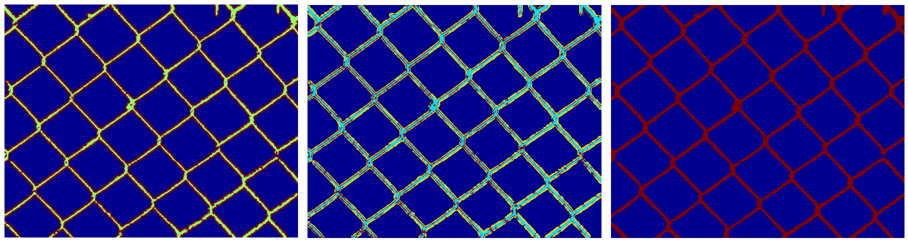
$$d(\mathrm{x}, \mathcal{G}) = \sum_{i=1}^{K} \pi_i \sqrt{(\mathrm{x} - \mu_i)^\top \Sigma_i^{-1} (\mathrm{x} - \mu_i)} \tag{4}$$

Any given pixel triple x is classified as an object pixel if its distance from the model is lower than a threshold τ, as a background pixel otherwise. Let Ω be the obtained object mask defined as:

$$\Omega(x, y) = \begin{cases} 1 & \text{if } d(I(x, y), \mathcal{G}) \leq \tau \\ 0 & \text{otherwise} \end{cases}$$

Here Ω represents the initial segmentation result. We observed that the initial segmentation may not be perfect; in particular, some non-target pixels may be classified as object pixels (false positives). Furthermore, some object regions, especially in the proximity of object boundaries, may be classified as background. We use morphological operators to eliminate the false positives and graph-cuts to include the false negatives. The false positives in the initial segmentation usually appear as isolated blobs. We use *open* ($\circ$) morphological operator to remove all the blobs with size less than γ.

Fig. 2 shows an image with user selected points on the target object, i.e. the fence. The GMM parameters and model order are iteratively estimated starting with $K_0 = 20$ clusters using the approach described in the previous section. Fig. 2b shows the value of $\mathcal{C}$ (Eq. 3) for different values of K: the minimum occurs at $K = 6$ which represents the optimal number of clusters for this example.



(a) (b) (c)

Fig. 3: (a) Trimap: Green color represents the F region, Blue color shows the B region and dark red represents the U region. (b) The U region is partitioned into F (shown in dark red) and B (shown in yellow) using graph-cuts, (c) Final segmentation results.

Fig. 2c shows the cluster wise distribution of the sample points and Fig. 2d shows the initial segmentation map Ω obtained with threshold $\tau = 4.7$. Small isolated islands of false positives can be seen in Fig. 2d whereas Fig. 2e shows the segmentation map after eliminating them using γ=30.

2.3 Segmentation Refinement via Graph-cuts

The false negatives usually appear in the proximity of the object boundaries and inside the segmentation map (see Fig. 2e). To improve the segmentation results we utilize the graph-cut techniques which have emerged as a powerful solution to many optimization problems in computer vision. A rough classification of the image pixels into at least two and ideally three classes is usually the input to a graph cut algorithm. Here we divide the image pixels into three classes namely, *definite foreground* F, *definite background* B and *unknown* U. The F and B classes are assumed as fixed and known whereas the pixels belonging to U class are to be decided upon. The image is then represented as an undirected weighted graph where the pixels form the vertices and edges link the vertices corresponding to neighbor pixels in the image. Two terminal nodes are added in the graph representing the F and B classes and they are linked to every other node in the graph. The edge weight represents the similarity between its end vertices and they are computed as described in [15]. Finally, the classification of U is performed by the iterative energy minimization through max-flow min-cut algorithm described in [16].

To define the trimap representing the corresponding classes: F, B and U, we take the segmentation map obtained from the previous section as F class. Since, the false negatives usually lie in and around the object boundary we use *dilation* ($\oplus$) morphological operator to estimate the 'Unknown' region U. The segmentation map Ω is dilated with structuring element s of size w:

$$\Omega_{dil} = \Omega \oplus s$$

Fig. 4: Segmentation of fences.

The unknown region U region is then computed as:

$$U = \Omega_{dil} \setminus \Omega$$

and region B is computed by negating the dilated mask:

$$B = \mathtt{NOT}(\Omega_{dil}) \tag{5}$$

Fig. 3a shows the trimap after dilation of mask (Fig. 2e) with structuring element of size 7×7. Fig. 3b shows the improvement achieved by graph-cut based refinement. Final segmented object is shown in Fig. 3c.

3 Experimental Evaluation and Results

The proposed segmentation technique is tested on images with a variety of distributed objects e.g. fences, bars, grills, wires, roads, rivers, etc. In each experiment the user marks few pixels on the target object. The number of user input points depends on the color characteristics of the target object. From experiment we found that 10 points are sufficient for objects with limited color variation whereas in case of large variation up to 20 clicks can be required. In all experiments the algorithm parameters are interactively tuned to optimize the segmentation results. In particular, κ=5 neighboring pixels are selected to increase the sample data and the structuring element size w was set to 7. The clustering is performed using CLUSTER library [32] and K_0 in model order estimation is set to 20.

Fig. 4 shows the segmentation of fence-like-objects from images. Fig. 5 shows the segmentation of similar objects scattered in the image. Segmentation of

Fig. 5: Segmentation of scattered objects. Left: a flock of sheep, middle: a bed of flowers, right: a picket fence.

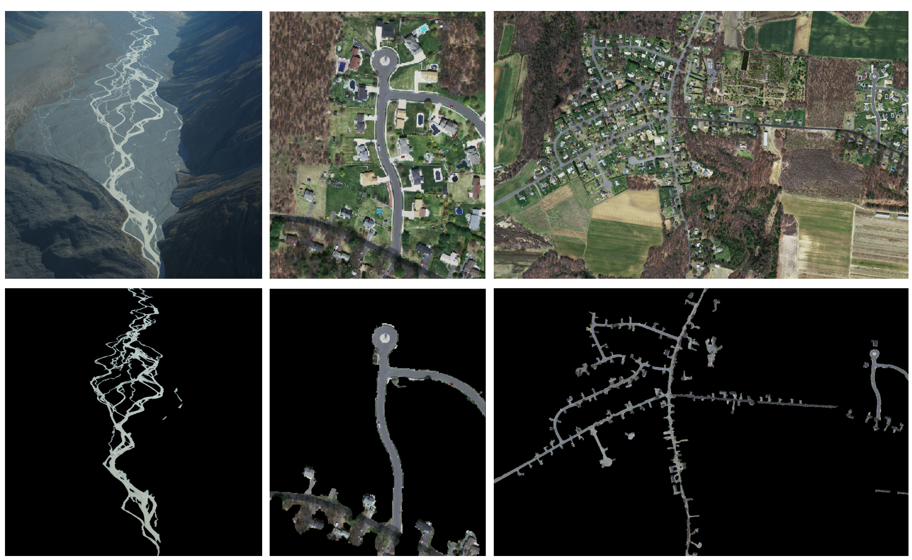


Fig. 6: Segmentation of river and road from aerial images. Left: segmentation of the Murchison river. Middle and right: segmentation of road from aerial images (the images are borrowed from Massachusetts Roads Dataset [31]).

roads and natural water sources e.g. rivers and streams from aerial images is an important problem in remote sensing. The proposed segmentation technique can be effectively used in such problems too. Fig. 6 shows the segmentation results of road and rivers from aerial images.

4 Conclusions

In this paper an object segmentation technique is proposed to segment objects which are scattered over the whole image e.g. fences, railings, roads, rivers. Since such objects are usually very thin and distributed over a large portion of the image, the conventional segmentation techniques are not effective. The proposed segmentation technique provides accurate and precise segmentation of such objects with minimum and easy user interaction. The obtained results on various distributed objects show the effectiveness of the proposed technique.

References

1. Osher, S., Sethian, J.: Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations. J. Comput. Phys. **79**(1) (1988) 12 – 49
2. Kuntimad, G., Ranganath, H.: Perfect image segmentation using pulse coupled neural networks. IEEE Trans. Neural Netw. **10**(3) (May 1999) 591–598
3. Beucher, S., Meyer, F.: The morphological approach to segmentation: the watershed transformation. Mathematical morphology in image processing. Opt. Eng. **34** (1993) 433–481
4. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Trans. Pattern Anal. Mach. Intell. **23**(11) (Nov 2001) 1222–1239
5. Ruzon, M., Tomasi, C.: Alpha estimation in natural images. In: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR). Volume 1. (2000) 18–25
6. Mubasher, M.M., Farid, M.S., Khaliq, A., Yousaf, M.M.: A parallel algorithm for change detection. In: 15th International Multitopic Conference (INMIC). (Dec 2012) 201–208
7. Farid, M.S., Mahmood, A., Grangetto, M.: Image de-fencing framework with hybrid inpainting algorithm. Signal Image Video Process. **10**(7) (2016) 1193–1201
8. Adobe Photoshop: Lasso tool. Available from http://helpx.adobe.com/photoshop/using/selecting-lasso-tools.html
9. Mortensen, E., Barrett, W.: Intelligent scissors for image composition. In: Proc. SIGGRAPH. (1995) 191–198
10. Mortensen, E., Morse, B., Barrett, W., Udupa, J.: Adaptive boundary detection using 'live-wire' two-dimensional dynamic programming. In: Proc. Comput. Cardiology. (Oct 1992) 635–638
11. Kang, H., Shin, S.: Enhanced lane: interactive image segmentation by incremental path map construction. Graph. Models **64**(5) (2002) 282 – 303
12. Kang, H.: G-wire: A livewire segmentation algorithm based on a generalized graph formulation. Pattern Recognit. Lett. **26**(13) (2005) 2042 – 2051

13. Ma, W.Y., Manjunath, B.: Edgeflow: a technique for boundary detection and image segmentation. IEEE Trans. Image Process. **9**(8) (Aug 2000) 1375–1388
14. Berman, A., Dadourian, A., Vlahos, P.: Method for removing from an image the background surrounding a selected object (October 17 2000) US Patent 6,134,346.
15. Boykov, Y., Funka-Lea, G.: Graph Cuts and Efficient N-D Image Segmentation. Int. J. Comput. Vis. **70**(2) (2006) 109–131
16. Rother, C., Kolmogorov, V., Blake, A.: "GrabCut": Interactive Foreground Extraction Using Iterated Graph Cuts. ACM Trans. Graph. **23**(3) (August 2004) 309–314
17. Peng, B., Zhang, L., Zhang, D.: A survey of graph theoretical approaches to image segmentation. Pattern Recognit. **46**(3) (2013) 1020 – 1038
18. Li, Y., Sun, J., Tang, C.K., Shum, H.Y.: Lazy snapping. ACM Trans. Graph. **23**(3) (August 2004) 303–308
19. Mortensen, E., Barrett, W.: Toboggan-based intelligent scissors with a four-parameter edge model. In: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR). Volume 2. (1999) 452–458
20. Vezhnevets, V., Konouchine, V.: Growcut: Interactive multi-label nd image segmentation by cellular automata. In: Proc. of Graphicon. (2005) 150–156
21. Juan, O., Boykov, Y.: Active Graph Cuts. In: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR). Volume 1. (June 2006) 1023–1029
22. Yang, Q., et al.: Progressive Cut: An image cutout algorithm that models user intentions. IEEE Multimedia **14**(3) (July 2007) 56–66
23. Peng, B., Zhang, L., Zhang, D., Yang, J.: Image segmentation by iterated region merging with localized graph cuts. Pattern Recognit. **44**(10–11) (2011) 2527 – 2538
24. Vicente, S., Kolmogorov, V., Rother, C.: Graph cut based image segmentation with connectivity priors. In: Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR). (June 2008) 1–8
25. Kolmogorov, V., Zabin, R.: What energy functions can be minimized via graph cuts? IEEE Trans. Pattern Anal. Mach. Intell. **26**(2) 147–159
26. Boykov, Y., Veksler, O.: Graph cuts in vision and graphics: Theories and applications. In: Handbook of Mathematical Models in Computer Vision. Springer US (2006) 79–96
27. Dempster, A., Laird, N., Rubin, D.: Maximum likelihood from incomplete data via the em algorithm. J. Royal Stat. Soc. Series B (Methodological) (1977) 1–38
28. Little, R.J.A., Rubin, D.B.: Statistical Analysis with Missing Data. 1st edn. Wiley Series in Probability and Statistics. Wiley, New York (1987)
29. Rissanen, J.: A universal prior for integers and estimation by minimum description length. Ann. Statist. **11**(2) (06 1983) 416–431
30. Mahalanobis, P.C.: On the generalised distance in statistics. Proc. National Institute of Science, India **2**(1) (April 1936) 49–55
31. Mnih, V.: Machine Learning for Aerial Image Labeling. PhD thesis, University of Toronto (2013)
32. Bouman, C.: Cluster: An unsupervised algorithm for modeling Gaussian mixtures. Available from http://engineering.purdue.edu/~bouman (April 1997)